

# The student's dilemma: ranking and improving prediction at test time without access to training data

Fabio Parisi<sup>1,#</sup>, Francesco Strino<sup>1,#</sup>, Boaz Nadler<sup>2</sup>, Yuval Kluger<sup>1,\*</sup>

**1** Yale University School of Medicine, Department of Pathology and Yale Cancer Center, 333 Cedar St., New Haven, CT 06520

**2** Weizmann Institute of Science, Department of Computer Science and Applied Mathematics, Rehovot, 76100 Israel

\* E-mail: yuval.kluger@yale.edu

# These authors contributed equally to this work.

## Abstract

The standard approach to rank the performance of several classifiers for a given classification problem is via an independent labeled validation dataset. However, in various applications only unlabeled data and several pre-constructed classifiers are provided, without access to labeled training or validation data. This begs the following questions: given only the predictions of several classifiers over a large set of unlabeled test data, is it possible to a) reliably rank their expected performances? and b) construct a meta-classifier more accurate than any individual classifier in the ensemble?

Here we present a spectral approach to address these questions. First, assuming errors of different classifiers are statistically independent, we show that the off-diagonal terms of their covariance matrix correspond to a rank-one matrix. Moreover, the entries of its leading eigenvector are proportional to the (balanced) accuracies of the classifiers. Second, using this eigenvector and without labeled data, we construct a novel spectral meta-learner (SML), which is a weighted linear combination of the classifiers in the ensemble. We interpret our SML as an approximation of the maximum likelihood estimator (MLE). Not only does SML typically achieve a higher accuracy than most classifiers in the ensemble, it also provides a better starting point for iterative estimation of the MLE than majority voting. Further, we show that SML is robust to the presence of small malicious groups of classifiers designed to veer the ensemble prediction away from the (unknown) ground truth. We demonstrate our unsupervised methods on several simulated and real datasets.

## Introduction

Imagine the following **student's dilemma**: a student is taking an exam, unprepared. However, during the test, the student gains access to the answers of fellow classmates. Expectedly, there is some disagreement between their answers. How should the student proceed to identify who, among the classmates, will get the highest grade? Is it possible for the student to cleverly combine the answers of his/her classmates and pass the exam with a grade better than all of them?

The first question above corresponds to the problem of estimating prediction performances of pre-constructed classifiers (e.g. fellow classmates) in absence of class labels. Namely, each classifier was constructed independently on a potentially different training dataset (e.g. each classmate studied on his/her own) and they are all being applied to a new test data  $D$  (e.g. the exam) for which labels are not available. In addition, the performance of each classifier on its own training data is unknown. This setting is markedly different from the typical supervised machine learning setting. There, classifiers are ranked after the class labels on the test dataset are disclosed in order to evaluate prediction performances. In the student's dilemma, classifiers are ranked based on an estimate of their prediction performance, inferred without any access to the class labels.

The second question may be addressed by a majority voting approach, which was used even in ancient times [1]. More recently this question was formulated as an iterative likelihood maximization procedure,

as exemplified by Dawid and Skene [2]. We note that if we had external knowledge or historical data to weigh the contribution of classifiers we could use other well-established approaches such as panels of experts [3, 4], or forecast combinations [5]; however, this knowledge is not available in the student’s dilemma and thus these solutions cannot be used to address our problem.

In recent years iterative likelihood maximization solutions were successfully applied to crowdsourcing problems, where multiple annotators with unknown degrees of expertise are requested to provide annotations of instances [6–13]. The focus of crowdsourcing is however different, since beyond the problem of inferring annotator’s accuracies, a major challenge is how to optimally decide on the number of annotators and how to assign instances to them. These problems do not arise in the student’s dilemma setting, where we assume that predicted labels can be obtained for all test data at virtually no cost from either human evaluators or algorithms/machine learning programs. Hence, our student’s dilemma setting can be seen as the full-data crowdsourcing case where all annotators provided predictions for all instances in the dataset.

In this paper we present four major contributions:

1. Under standard independence assumptions between classifier errors, we prove that in the limit of an infinite test set, the off-diagonal entries of the population covariance matrix of all classifiers correspond to a rank-one matrix.
2. We show that the entries of the first eigenvector of this rank-one matrix are proportional to the balanced accuracies of the classifiers. Thus, a spectral decomposition of this rank-one matrix provides a fast approach to sort the performances of an ensemble of classifiers. To the best of our knowledge, this gives the first computationally efficient and asymptotically consistent solution to the classical problem posed by Dawid and Skene [2] in 1979, for which thus far only non-convex iterative likelihood maximization solutions have been proposed [8, 14–17].
3. We propose the Spectral Meta-Learner (SML): A new, easy to construct unsupervised ensemble-learner. Not only does SML typically performs better than most classifiers in the ensemble or their majority vote, it is also a better starting point for maximum likelihood estimation (MLE) using the expectation-maximization (EM) algorithm.
4. We show that SML is robust to the presence of conspiring classifiers (representing a cartel or an interest group), which maliciously attempt to veer the overall ensemble solution away from the (unknown) ground truth.

## 1 Problem setup

Let  $\{f_i\}_{i=1}^M$  be  $M$  binary classifiers, whose inputs belong to some instance space  $\mathcal{X}$  (typically  $\mathcal{X} = \mathbb{R}^d$ ). We assume that each classifier  $f_i$  was trained individually in a manner undisclosed to us using its own labeled training set, which is also unavailable to us. Thus, we view each classifier as a *black-box function*  $f_i : \mathcal{X} \rightarrow \{-1, 1\}$  with an unknown classification performance.

Let  $D = \{x_k\}_{k=1}^S \subset \mathcal{X}$  be a test set of  $S$  unlabeled samples,  $\mathbf{y} = (y_1, \dots, y_S)$  be their true (unknown) labels, and  $f_i(x_k)$  be the label predicted by the  $i$ -th classifier at  $x_k$ .

Given only the predictions of the  $M$  classifiers on the unlabeled set  $D$  and no other labeled data, we consider the following two problems: i) *rank* the performances of the  $M$  classifiers; and ii) construct an improved estimate  $\hat{\mathbf{y}} = (\hat{y}_1, \dots, \hat{y}_S)$  of the label vector  $\mathbf{y}$ .

## 2 Ranking of classifiers

We first introduce some notation and state our assumptions. Let  $(X, Y) \in \mathcal{X} \times \{-1, 1\}$  be a random vector corresponding to our binary classification problem,  $p(x, y)$  its probability density function, and

$p(x)$  the marginal of  $X$ .

In the present study, we measure the performance of a binary classifier  $f$  by its **balanced accuracy**  $\pi$ , defined as

$$\pi = \frac{\text{sensitivity} + \text{specificity}}{2} = \frac{1}{2}(\psi + \eta) \quad (1)$$

where  $\psi$  and  $\eta$  are the sensitivity and specificity, respectively, of the classifier  $f$ ,

$$\psi = \Pr[f(X)=Y|Y=1], \text{ and } \eta = \Pr[f(X)=Y|Y=-1] \quad (2)$$

Balanced accuracy is a common measure of quality of a classifier, in particular when one class label is much more abundant than the other. As discussed below, in our setting balanced accuracy arises as the natural measure to consider.

In our analysis we make the following two assumptions: i) The  $S$  unlabeled samples  $x_k \in D$  are i.i.d. realizations from the marginal distribution  $p(x)$ ; and ii) the  $M$  classifiers are statistically independent, in the sense that prediction errors made by one classifier are independent of those made by any of the other classifiers. Namely, for all  $1 \leq i \neq j \leq M$ , and for each of the two class labels, with  $y, a_i, a_j \in \{-1, 1\}$

$$\Pr[f_i(X)=a_i, f_j(X)=a_j|Y] = \Pr[f_i(X)=a_i|Y] \Pr[f_j(X)=a_j|Y]. \quad (3)$$

Note that these assumptions are standard both in the development of supervised ensemble methods [18], as well as in other works considering a setting similar to ours [2, 13].

To understand how we may rank the classifiers without labeled data, it is instructive to consider the population setting, whereby the number of unlabeled test data tends to infinity,  $|D| = S \rightarrow \infty$ . Let  $Q$  be the  $M \times M$  population covariance matrix of the  $M$  classifiers, with entries

$$q_{ij} = \mathbb{E}[(f_i(X) - \mu_i)(f_j(X) - \mu_j)] \quad (4)$$

where  $\mathbb{E}$  denotes expectation with respect to the density  $p(x, y)$  and  $\mu_i = \mathbb{E}[f_i(X)]$ .

The following lemma, proved in the supplementary information, characterizes the relation between the matrix  $Q$  and the balanced accuracies of the  $M$  classifiers:

**Lemma 2.1.** *The entries  $q_{ij}$  of  $Q$  are given by*

$$q_{ij} = \begin{cases} 1 - \mu_i^2 & i = j \\ (2\pi_i - 1)(2\pi_j - 1)(1 - b^2) & \text{otherwise} \end{cases} \quad (5)$$

where  $b \in (-1, 1)$  is the class imbalance,

$$b = \Pr[Y = 1] - \Pr[Y = -1]. \quad (6)$$

The key insight from this lemma is that the off-diagonal entries of  $Q$  are identical to those of a *rank-one matrix*  $\tilde{Q} = \lambda \mathbf{v} \mathbf{v}^T$  with unit-norm eigenvector  $\mathbf{v}$  and eigenvalue

$$\lambda = (1 - b^2) \cdot \sum_{i=1}^M (2\pi_i - 1)^2 \quad (7)$$

Importantly, up to a sign ambiguity, the entries of  $\mathbf{v}$  are *proportional* to the balanced accuracies,

$$v_i \propto (2\pi_i - 1). \quad (8)$$

Hence, the  $M$  classifiers can be ranked according to their balanced accuracies by sorting the entries of the eigenvector  $\mathbf{v}$ .

In practice, neither  $Q$  nor  $\mathbf{v}$  are known, but both can be estimated from the finite unlabeled dataset  $D$ . We denote the corresponding sample covariance matrix by  $\hat{Q}$ . Its entries are

$$\hat{q}_{ij} = \frac{1}{S-1} \sum_{k=1}^S (f_i(x_k) - \hat{\mu}_i)(f_j(x_k) - \hat{\mu}_j)$$

where  $\hat{\mu}_i = \frac{1}{S} \sum_k f_i(x_k)$ . Under our assumptions,  $\hat{Q}$  is an unbiased estimate of  $Q$ , namely  $\mathbb{E}[\hat{Q}] = Q$ . Moreover, the variances of the off-diagonal entries of the sample covariance matrix  $\hat{Q}$  are

$$\text{Var}[\hat{q}_{ij}] = \frac{(1 - \mu_i^2) \cdot (1 - \mu_j^2)}{S-1} + \frac{q_{ij}}{S} \left( 4\mu_i\mu_j - \frac{S-2}{S-1} \hat{q}_{ij} \right). \quad (9)$$

Finally,  $\hat{Q} \rightarrow Q$  as  $S \rightarrow \infty$ . Hence, for a sufficiently large unlabeled set  $D$ , it should be possible to accurately estimate the ranking of the  $M$  classifiers from  $\hat{Q}$ .

In fact, the discussion above suggests several ways to rank the  $M$  classifiers. One option is to look for a rank-one matrix  $R = \hat{\lambda} \hat{\mathbf{v}} \hat{\mathbf{v}}^T$ , whose off-diagonal terms are closest to those of  $\hat{Q}$ . While the rank-one constraint is non-convex, its standard relaxation to a trace constraint yields

$$\hat{R} = \arg \min \sum_{i \neq j} (\hat{q}_{ij} - R_{ij})^2 + \theta \text{Trace}(R) \quad (10)$$

subject to  $R = R^T$ , and  $R \succeq 0$ . This is a convex problem, which can be solved efficiently (in polynomial time in  $M$ ) via semi-definite programming [19].

An alternative and more computationally efficient approach is to construct an estimator of  $\tilde{Q}$ , and then compute its leading eigenvector  $\hat{\mathbf{v}}$ . Given that  $\mathbb{E}[\hat{Q}] = Q$ , we estimate the off-diagonal entries of  $\tilde{Q}$  by those of  $\hat{Q}$ . As for the diagonal entries, note that upon the change of variables  $\tilde{q}_{ii} = e^{t_i}$ , for all  $i \neq j$

$$\log |\hat{q}_{ij}| - t_i - t_j = 0.$$

In the finite sample setting, we replace the unknown  $q_{ij}$  by  $\hat{q}_{ij}$  and look for an  $M$ -dimensional vector  $\mathbf{t}$  such that the relation above holds approximately for all pairs  $i \neq j$ ,

$$\hat{\mathbf{t}} = \arg \min \sum_{j > i} (\log |\hat{q}_{ij}| - \hat{t}_i - \hat{t}_j)^2 \quad (11)$$

The vector  $\hat{\mathbf{t}}$  is efficiently found by solving an  $M \times M$  system of linear equations. Since  $\hat{q}_{ij} \rightarrow q_{ij}$  as  $S \rightarrow \infty$ , it follows that  $\hat{\mathbf{t}}$  is an asymptotically consistent estimate of  $\mathbf{t}$ , and consequently the resulting  $\hat{\mathbf{v}}$  is a consistent estimate of  $\mathbf{v}$ .

In practice, to avoid the singularity at zero of the logarithm function, we modify (11) by summing only over indices  $i, j$  for which  $|\hat{q}_{ij}| > 2\sqrt{\text{Var}[\hat{q}_{ij}]}$ , where  $\text{Var}[\hat{q}_{ij}]$  is a plug-in estimator of (9). Once  $\hat{\mathbf{t}}$  is found, we construct the estimate of  $\tilde{Q}$  and rank the  $M$  classifiers by its leading eigenvector  $\hat{\mathbf{v}}$ .

Finally, an even simpler approach is to rank the classifiers by directly computing the leading eigenvector of  $\hat{Q}$ . For a finite number of classifiers  $M$ , it follows from Lemma 2.1 that as  $S \rightarrow \infty$  this approach is in general *not* consistent. However, as the following lemma shows, if  $M$  is large this leading eigenvector is close to the true one.

**Lemma 2.2.** *Let  $\mathbf{w}$  be the leading unit-norm eigenvector of the population matrix  $Q$ , and let  $\lambda$  be given by (7). Then,*

$$(\mathbf{w}^T \mathbf{v})^2 \geq 1 - \frac{2}{\lambda} \quad (12)$$

A proof of Lemma 2.2 is provided in the Supplementary Information. Note that if all classifiers in the ensemble have a balanced accuracy bounded away from  $1/2$ , then  $\lambda = O(M)$  and the angle between  $\mathbf{v}$  and  $\mathbf{w}$  is small.

Ranking classifiers by a singular value decomposition of the  $S \times M$  matrix of predicted labels  $f_i(x_k)$  was recently suggested in [6], where the  $j$ -th entry in the leading right singular vector was considered a proxy for the reliability of the  $j$ -th classifier. Our work provides a novel probabilistic interpretation to their approach, as it shows that the entries of  $\mathbf{w}$  (also the leading right singular vector of the matrix  $f_i(x_k)$ ) are approximately those of  $\mathbf{v}$ , which in turn are proportional to the balanced accuracies of the classifiers. Consistent with the analysis above, in our simulations we found that all three approaches (SDP (10), least-squares problem (11) and direct eigen-decomposition of  $Q$ ) gave comparable rankings, though the latter was slightly less accurate.

### 3 The Spectral Meta Learner (SML)

Next, we turn to the problem of constructing a meta-learner expected to be more accurate than any of the  $M$  classifiers in the ensemble. In our setting, this is equivalent to estimating the  $S$  unknown labels  $y_1, \dots, y_S$  by combining the labels predicted by the  $M$  classifiers.

The standard approach to this task is to determine for all the unlabeled instances the maximum likelihood estimator (MLE)  $\hat{\mathbf{y}}^{\text{ML}}$  of their true class labels  $\mathbf{y}$  [2]. Under the assumption of independence between classifier errors and independence between instances, the overall likelihood is the product of the likelihoods of the  $S$  individual instances, where the likelihood of a label  $y$  for an instance  $x$  is

$$\mathfrak{L}(f_1(x), \dots, f_M(x); y) = \prod_{i=1}^M \Pr[f_i(x)|y]. \quad (13)$$

As shown in the Supplemental Information, the MLE can be written as a weighted sum of the binary labels  $f_i(x) \in \{-1, 1\}$ , with weights that depend on the sensitivities  $\psi_i$  and specificities  $\eta_i$  of the classifiers. For an instance  $x$ ,

$$\begin{aligned} \hat{y}^{(\text{ML})} &= \underset{y}{\operatorname{argmax}} \ln \mathfrak{L}(f_1(x), \dots, f_M(x); y) \\ &= \operatorname{sign} \left( \sum_{i=1}^M f_i(x) \log \alpha_i + \log \beta_i \right) \end{aligned} \quad (14)$$

with

$$\alpha_i = \frac{\psi_i \eta_i}{(1 - \psi_i)(1 - \eta_i)}, \quad \beta_i = \frac{\psi_i(1 - \psi_i)}{\eta_i(1 - \eta_i)}. \quad (15)$$

Equation (14) shows that the MLE is a *linear ensemble classifier*, whose weights depend, unfortunately, on the unknown specificities and sensitivities of the  $M$  classifiers.

The common approach, pioneered by Dawid and Skene [2], is to *jointly* maximize the likelihood of all  $S$  labels and the specificities and sensitivities of the  $M$  classifiers. Given an estimate of the true class labels, it is straightforward to estimate each classifier sensitivity and specificity. Similarly, given estimates of  $\psi_i$  and  $\eta_i$ , the corresponding estimates of  $\mathbf{y}$  are easily found via (14). Hence, the MLE is typically approximated by expectation-maximization (EM) [8–11, 13].

As is well known, the EM procedure is guaranteed to increase the likelihood at each iteration. However, its key limitation is that since the likelihood is in general a non-convex function, the EM iterations may converge to a local (rather than global) maximum of the likelihood function.

Importantly, the EM procedure requires an initial guess of the ground truth labels  $\mathbf{y}$ . A common choice is the simple majority voting rule of the ensemble of classifiers. As noted in previous studies, majority voting may be highly suboptimal, and starting the EM procedure from it may lead to suboptimal local maxima [13]. Thus, it is desirable, and as described below in some cases crucial, to initialize the EM algorithm with an estimate  $\hat{\mathbf{y}}$  that is close to the true class label  $\mathbf{y}$ .

In this section we show that it is indeed possible to construct a more accurate initial guess, using the eigenvector of the previous section. To this end we note that a Taylor expansion of the unknown coefficients  $\alpha_i$  and  $\beta_i$  in (15) around  $(\psi_i, \eta_i) = (1/2, 1/2)$  gives, up to second order terms  $O((\psi_i - 1/2)^2, (\eta_i - 1/2)^2, (\psi_i - 1/2) \cdot (\eta_i - 1/2))$ ,

$$\alpha_i \approx 1 + 4(\psi_i + \eta_i - 1), \quad \beta_i \approx 1. \quad (16)$$

Next, recall that the balanced accuracy is  $\pi_i = (\eta_i + \psi_i)/2$ . Hence, combining a Taylor expansion of (14) around  $(\psi_i, \eta_i) = (1/2, 1/2)$  with (16) and keeping only first order terms yields

$$\hat{y}_k^{(\text{ML})} \approx \text{sign} \left( \sum_{i=1}^M f_i(x_k)(2\pi_i - 1) \right). \quad (17)$$

Recall that by Lemma 2.1, up to a sign ambiguity the entries of the first eigenvector of  $\tilde{Q}$  are proportional to the balanced accuracies of the classifiers,  $v_i \propto (2\pi_i - 1)$ . This sign ambiguity can be easily removed if we assume, for example, that most classifiers are better than random. Replacing  $2\pi_i - 1$  in (17) by the eigenvector entries  $\hat{v}_i$  of an estimate of  $\tilde{Q}$  yields a novel spectral-based ensemble classifier, which we term the Spectral Meta Learner (SML),

$$\hat{y}_k^{(\text{SML})} = \text{sign} \left( \sum_{i=1}^M f_i(x_k) \cdot \hat{v}_i \right). \quad (18)$$

As we shall see in the simulation section, the SML is typically more accurate than majority voting, and provides a better initial guess for EM procedures that estimate the MLE.

## 4 Learning in the Presence of a Malicious Cartel

We now consider a scenario whereby for some  $r \in [0, 1/2)$ ,  $r \cdot M$  classifiers belong to a conspiring **cartel** (e.g. representing a junta or an interest group), maliciously designed to veer the ensemble solution toward the cartel's target and away from the truth. The possibility of such a scenario raises the following question: how sensitive are SML and majority voting to the presence of a cartel? In other words, to what extent can these methods remove, or at least substantially reduce the effect of the cartel classifiers, without knowing their identity or applying sophisticated clustering algorithms to identify them.

To this end, let us first introduce some notation. Let the ensemble of  $M$  classifiers be composed of a set  $P$  of  $(1-r)M$  "honest" classifiers and a set  $C$  of  $rM$  malicious cartel classifiers. The honest classifiers satisfy the assumptions of the previous section: each classifier attempts to correctly predict the truth with a balanced accuracy  $\pi_i$ , and different classifiers make independent errors. The cartel classifiers, in contrast, attempt to predict a *different* target labeling,  $\mathbf{T}$ . We assume that conditional on both the cartel's target and the true label, the classifiers in the cartel make independent errors. Namely, for all  $i, j \in C$ , and for any labels  $a_i, a_j \in \{-1, 1\}$

$$\Pr[f_i(X) = a_i, f_j(X) = a_j | T, Y] = \Pr[f_i(X) = a_i | T] \Pr[f_j(X) = a_j | T]. \quad (19)$$

Finally, we assume that the prediction errors of cartel and honest classifiers are also independent.

The following lemma, proven in the supplementary information, characterizes the relation between the population matrix  $Q$  and the following quantities: the balanced accuracies of the  $M$  classifiers, the balanced accuracy  $\pi_c$  of the cartel's target with respect to the truth, and the balanced accuracies  $\xi_j$  of the  $r \cdot M$  cartel members relative to their target:

**Lemma 4.1.** *Given  $(1-r)M$  honest classifiers and  $r \cdot M$  classifiers of a cartel  $C$ , the entries  $q_{ij}$  of  $Q$  satisfy*

$$q_{ij} = \begin{cases} 1 - \mu_i^2 & i = j \\ (2\pi_i - 1)(2\pi_j - 1)(1 - b^2) & i \in P, j \in P \\ (2\pi_i - 1)(2\pi_c - 1)(2\xi_j - 1)(1 - b^2) & i \in P, j \in C \\ (2\xi_i - 1)(2\xi_j - 1)(1 - b^2) & i \in C, j \in C \end{cases} \quad (20)$$

where  $b \in (-1, 1)$  is the class imbalance, as in (6).

Based on Lemma 4.1, the following theorem shows that in the presence of a single cartel, the off-diagonal entries of  $Q$  correspond to a *rank-two* matrix. We conjecture that in the presence of  $k$  independent cartels, the respective rank is  $(k+1)$ .

**Theorem 4.2.** *Given  $(1-r)M$  honest classifiers and  $rM$  classifiers belonging to a cartel,  $0 < r < 1$ , the off-diagonal entries of  $Q$  correspond to a **rank-two matrix** with eigenvalues*

$$\begin{aligned} \lambda_1 &= \lambda_P \cos^2 \alpha + \lambda_C \sin^2 \beta \\ \lambda_2 &= \lambda_P \sin^2 \alpha + \lambda_C \cos^2 \beta \end{aligned} \quad (21)$$

and eigenvectors

$$e_{1i} = \begin{cases} (2\pi_i - 1) \cos \alpha & i \in P \\ (2\xi_i - 1) \sin \beta & i \in C \end{cases} \quad (22)$$

$$e_{2i} = \begin{cases} (2\pi_i - 1) \sin \alpha & i \in P \\ (2\xi_i - 1) \cos \beta & i \in C \end{cases}, \quad (23)$$

where

$$\lambda_P = (1 - b^2) \sum_{j \in P} (2\pi_j - 1)^2, \quad \lambda_C = (1 - b^2) \sum_{j \in C} (2\xi_j - 1)^2. \quad (24)$$

and

$$\alpha = \frac{1}{2} \arctan \left( \frac{k_1 k_2}{k_2(1 - 2k_1^2) - 1} \right), \quad \beta = \frac{1}{2} \arctan \left( \frac{2k_1 \sqrt{1 - k_1^2}}{1 - k_2 - 2k_1^2} \right) \quad (25)$$

$$k_1 = 2\pi_c - 1, \quad k_2 = \lambda_C / \lambda_P \quad (26)$$

An intuitive interpretation of Theorem 4.2 is that the covariance matrix  $Q$  describes a two-dimensional subspace. The honest classifiers lie on a line with angle  $\alpha$  relative to the eigenvector  $e_1$ . The cartel classifiers lie on a line with angle  $\beta$  relative to the eigenvector  $e_2$ .

As an illustrative example, we consider the case where the cartel's target is uninformative with respect to the truth, i.e.  $\pi_c = 1/2$ . In this case  $\alpha = \beta = 0$ , so  $\lambda_1 = \lambda_P$ ,  $\lambda_2 = \lambda_C$  and

$$e_{1i} = \begin{cases} 2\pi_i - 1 & i \in P \\ 0 & i \in C \end{cases} \quad (27)$$

$$e_{2i} = \begin{cases} 0 & i \in P \\ 2\xi_i - 1 & i \in C \end{cases} \quad (28)$$

Recall from (18) that SML weighs each classifier by the corresponding entry in the leading eigenvector. Hence, if the cartel’s target is orthogonal to the truth ( $\pi_c = 1/2$ ) and  $\lambda_P > \lambda_C$ , SML asymptotically *ignores* the cartel (Fig. S1). In contrast, regardless of  $\pi_c$ , majority voting is affected by the cartel, proportionally to its fraction size  $r$ . Hence, SML is much more robust than majority voting to the presence of a cartel.

## 5 Results

This section contains two parts. First, we study our ranking and SML algorithms on simulated data for both an ensemble of independent classifiers, and an ensemble of independent classifiers corrupted by the presence of one cartel. In the second part, using standard machine learning algorithms as our collection of binary classifiers, we evaluate our methods on several real datasets from medical, biological and engineering applications. This second part shows that our methods are robust to deviations from the (unrealistic) strict independence assumptions on errors between classifiers.

### 5.1 Simulated Data

In our simulations we considered an unlabeled test data of size  $S = 600$  instances, a ground truth with class imbalance  $b = 0$  and an ensemble of  $M = 100$  classifiers. Each classifier had potentially different sensitivity and specificity chosen at random such that its balanced accuracy was uniformly distributed on the interval  $[0.3, 0.8]$ . This setup was chosen to imitate a difficult learning problem with independent classifiers, some of which are worse than random. We note that classifiers that are worse than random may occur in real studies, where the training data is too small in size or not sufficiently representative of the test data. Finally, we considered the effect of a malicious cartel consisting of 33% of the classifiers, having their own target labeling. More details about the simulations are provided in the supplementary information.

*Ranking of Classifiers:* We constructed the sample covariance matrix, corrected its diagonal according to (11) and computed its leading eigenvector  $\hat{\mathbf{v}}$ . In both cases (independent classifiers and cartel), with probability of at least 80%, the classifier with highest accuracy was also the one with the largest entry (in absolute value) in the eigenvector  $\hat{\mathbf{v}}$ , and with probability  $> 99\%$  its inferred rank was among the top five classifiers (Fig. S2). We remark that even if the test data of size  $S = 600$  were fully labeled, identifying the best performing classifier would still be prone to errors, since the estimated balanced accuracy has itself an error of  $O(1/\sqrt{S})$ .

*Unsupervised Ensemble-Learning:* Next, for the same set of simulations we compared the balanced accuracy of majority voting and of our suggested SML. We also considered the predictions of these two meta-learners as starting points for iterative EM calculation of the MLE (iMLE). As shown in Fig. 1, SML was significantly more accurate than majority voting. Furthermore, applying an EM procedure with SML as an initial guess provided relatively small improvements in the balanced accuracy. Majority voting, in contrast, was less robust. Moreover, in the presence of a cartel, computing the MLE with majority voting as its starting point exhibited a multi-modal behavior, sometimes converging to a local maxima with a relatively low balanced accuracy.

A more detailed study of the sensitivity of SML and majority voting and their respective improved iMLE solutions to the size of a malicious cartel with  $\pi_c = 0.5$  is shown in Fig. 2. As expected, the average balanced accuracy of SML, voting or iMLE initialized using either voting or SML decreases as a function of the cartel’s fraction,  $r$ , and once the cartel’s fraction is too large all methods fail. In our simulations, both SML and iMLE initialized with it were far more robust to the size of the cartel, in comparison to both majority voting and iMLE initialized with it. With a cartel size of 20%, SML was still able to construct a nearly perfect predictor, whereas the balanced accuracy of majority voting and



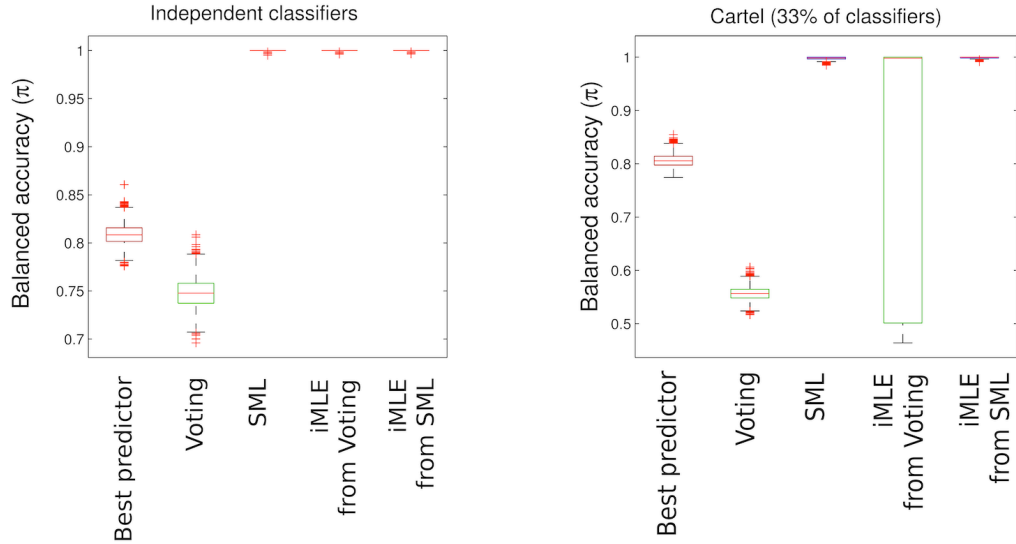


Figure 1: The Spectral Meta Learner (SML) is a good and robust meta-learner. The performance of SML is higher than that of majority voting (green) also in the presence of one cartel. In the presence of a cartel with target balanced accuracy of 0.5 (right panel), iMLE initialized with SML benefits from its robustness to cartels. In contrast, iMLE initialized using majority voting may converge to a poor local maxima. The boxplots represent the distribution of balanced accuracies of 3000 independent runs.

iMLE initialized with it were both far from 1. Interestingly, the prediction of iMLE using SML as starting condition showed no significant improvement relative to the average balanced accuracy of SML itself.

## 5.2 Real Datasets

We applied our spectral approach to 8 common and publicly available datasets from several scientific and engineering applications. We used 33 standard machine-learning methods implemented in the software package Weka [32] as our suite of classifiers. Details on the datasets and the classifiers used appear in the Supplementary Information (Table S1) and Table S2).

We split each dataset into a labeled part, and an unlabeled part, the latter serving as the test data  $D$  used to evaluate our methods. To best reproduce the problem setting of the student’s dilemma, each algorithm had access only to a subset of the labeled data (i.e. each classifier was trained with a slightly different training set). For each of these eight datasets, the leading eigenvector of the modified covariance matrix was highly concordant with the classifier’s balanced accuracies computed on the test set after disclosure of the true class labels, regardless of potential dependencies between them, with a Kendall’s  $\tau$  correlation typically higher than 0.9. One exception was the Abalone dataset, for which all classifiers had poor accuracy and hence were difficult to rank.

Next, we compared SML, majority voting and iMLE initialized at either of these two classifiers. As seen in Fig. 3, consistently across all datasets, iMLE initialized with SML had a higher mean balanced accuracy than iMLE initialized with majority voting. Furthermore, iMLE initialized with SML was more robust, with fewer outliers having low balanced accuracy.

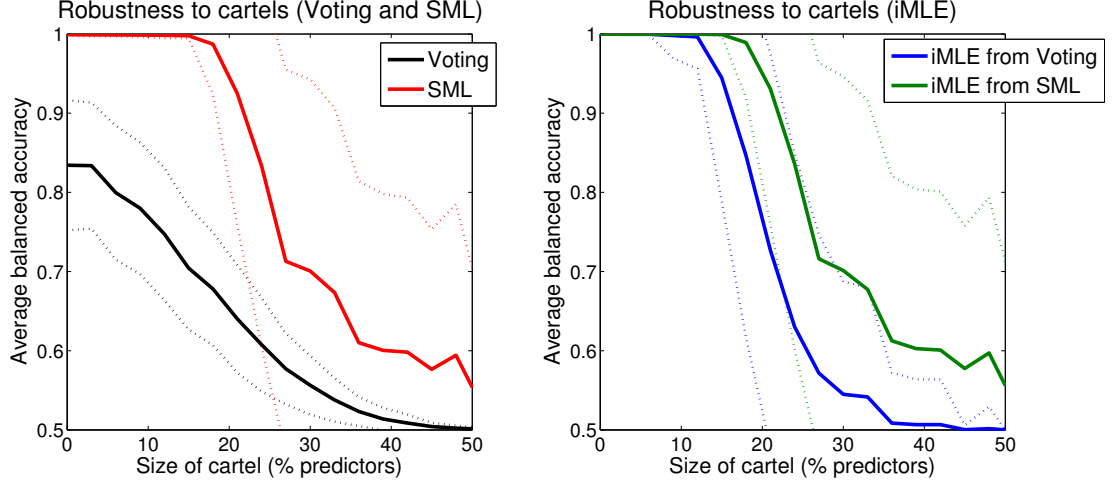


Figure 2: SML is more robust to cartels than majority voting (left panel). iMLE using SML estimates as starting point is also more robust to cartels than iMLE using majority voting as the starting condition (right panel). For each meta-learner prediction the average balanced accuracy is shown (filled lines) together with the standard error (dotted lines,  $n=500$  runs for each cartel’s fraction).

## 6 Summary and Discussion

In the present work, we developed an unsupervised spectral framework to rank the performances of binary classifiers and to combine their predictions into an ensemble spectral meta-learner, SML, that is easy to construct and fast to compute. We showed that SML is equivalent to linearization of the MLE around  $(\psi, \eta) = (1/2, 1/2)$ . This is the only neighborhood where linearization of MLE is invariant to substitution of the unknown balanced accuracies by the corresponding entries of the eigenvector  $\mathbf{v}$ . Interestingly, we found that in most cases the prediction returned by iMLE starting from SML is only slightly better than the prediction obtained by SML itself, suggesting that the SML solution nearly coincides with a local maximum of the likelihood function. In addition, we showed that SML is robust to cartels. Finally, we illustrated the applicability of the proposed methods on data from real-world problems.

Our work raises several interesting problems for future research. First, most of our analysis was asymptotic, in the limit of an infinitely large unlabeled test set. A theoretical study of the effects of a finite test set on the accuracy of the leading eigenvector are of interest. This is particularly relevant in the crowdsourcing setting, where there is significant missing data in the prediction matrix  $f_i(x_k)$ . In principle, an estimated covariance matrix can be computed by using the complete observations for each pair of classifiers. However, perhaps an alternative approach of directly fitting a low rank matrix is more suitable.

A natural extension of the present work is to analyze problems where the response class label is categorical rather than binary (multi-class problems), or even continuous (regression problems). We expect that even in these problems the covariance matrix of the predictions of independent classifiers (or independent regressors) is a perturbation of a low-rank matrix. A modified covariance matrix, similar to the one proposed in our study, may improve the quality of existing methods.

The quality of predictions may also be improved by taking into consideration instance difficulty, discussed in previous studies [8, 13]. In these studies there is an assumption that some instances are harder to classify correctly, independent of the classifier employed, with different analytic formulations proposed to model this difficulty. In our context, both very easy examples (on which all classifiers agree) and very difficult ones (on which classifier predictions are as good as random) are not useful for ranking

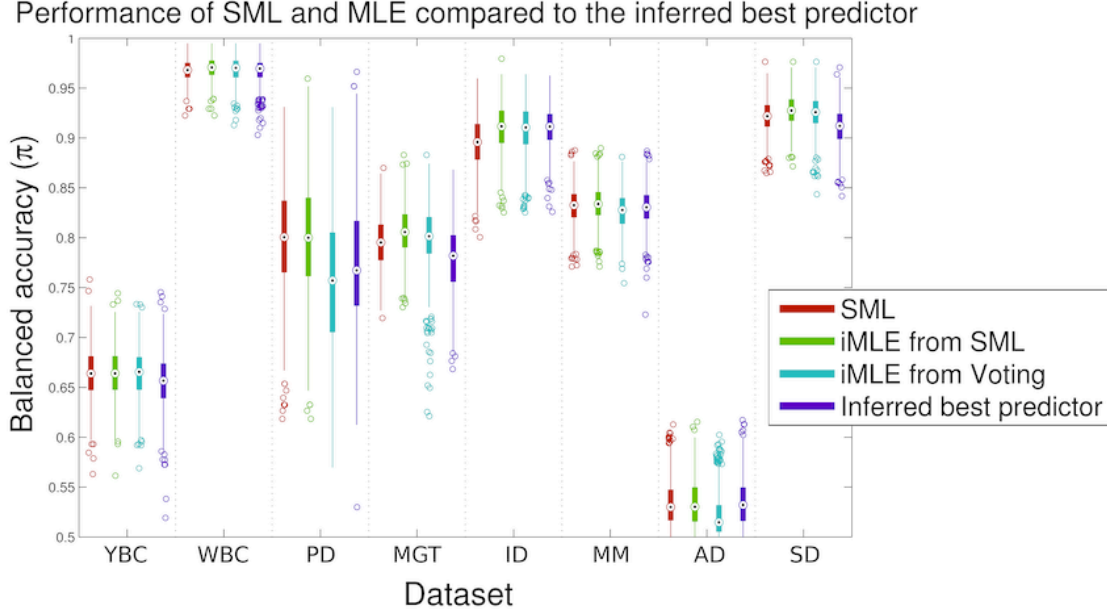


Figure 3: Comparison of several classifiers on all eight datasets. Compared to MLE from voting, SML and MLE from it were overall more robust with fewer cases of low balanced accuracy, and in some datasets (PD, AD) achieved a significant higher median balanced accuracy. For each dataset, the boxplots represent the distribution of balanced accuracies across 1000 independent runs.

the different classifiers. This suggests that in the presence of instance difficulty, if it can somehow be estimated, then it may be profitable to rank the classifiers by stratifying the data and removing these very easy or very hard samples. On the theoretical front, incorporating instance difficulty into our model may require additional and more restrictive assumptions concerning the independence between classifiers and between instances, for example at each difficulty level.

The current formulation provides no measure of the confidence of class-label assignment using SML. A relaxation of (18) obtained by considering the argument of the sign operator can be used to assess the confidence of the class assignment of each instance. This formulation can be used with performance measures such as the Area Under the Receiver Operator Characteristic Curve.

In the present work we also introduced the notion of cartels. The ability to identify such groups and their target, as well as to ignore their contributions, is of critical importance in many practical applications, such as electoral committees and decision-making in trading. We showed how the SML prediction asymptotically ignores moderately sized cartels. We conjecture that such construction is possible for  $\pi_c \approx 1/2$  even when the honest predictors are a minority. In this scenario  $\lambda_C > \lambda_P$ , thus the SML prediction should be constructed using the eigenvector associated to the second eigenvalue,  $\lambda_P$  in this case.

## Materials and Methods

### Datasets and Classifiers

In the present study we used 8 datasets for binary classification problems. With the exception of the Yale breast cancer dataset [22], these datasets were obtained from the public ICS repository [23]. Details

on each dataset are provided in the Supplemental Information. The classifiers used in the present study have been previously described [24] or have been implemented in the software package Weka [32].

## **Statistical Analysis and Visualization**

Statistical analysis and visualization of results have been performed using MATLAB (2012a, The Math-Works, Natick, MA). Visualization of distributions has been performed using boxplots [25].

## **Acknowledgments**

We thank Amit Singer, Alex Kovner, Ronald Coifman, Ronen Basri and Joseph Chang for their invaluable feedback and encouragement. The Wisconsin breast cancer dataset is based on data collected at the University of Wisconsin Hospitals, Madison by Dr. William H. Wolberg and colleagues. F.S. is supported by the American-Italian Cancer Foundation [Post-Doctoral Research Fellowship to F.S.].

# Supplementary Information: The student's dilemma: ranking and improving prediction at test time without access to training data

Fabio Parisi<sup>1, #</sup>, Francesco Strino<sup>1, #</sup>, Boaz Nadler<sup>2</sup>, Yuval Kluger<sup>1, \*</sup>

## 1 Covariance between different classifiers

*Proof of Lemma 2.1.* To prove the lemma we first compute the mean  $\mu_i = \mathbb{E}[f_i(X)]$  and the variance  $\text{Var}[f_i(X)]$  of the  $i$ -th classifier. We then use these results to compute the entries of the population covariance matrix,  $q_{ij} = \mathbb{E}[(f_i(X) - \mu_i) \cdot (f_j(X) - \mu_j)]$ .

Under the assumption of independence between instances, the population mean  $\mu_i = \mathbb{E}[f_i(X)]$  of the  $i$ -th classifier is

$$\begin{aligned} \mathbb{E}[f_i(X)] &= \Pr[f_i(X) = 1] - \Pr[f_i(X) = -1] \\ &= \Pr[f_i(X) = 1|Y = 1] \Pr[Y = 1] + \Pr[f_i(X) = 1|Y = -1] \Pr[Y = -1] \\ &\quad - \Pr[f_i(X) = -1|Y = 1] \Pr[Y = 1] - \Pr[f_i(X) = -1|Y = -1] \Pr[Y = -1] \end{aligned} \quad (29)$$

Using the definitions of sensitivity  $\psi_i = \Pr[f_i(X) = 1|Y = 1]$ , specificity  $\eta_i = \Pr[f_i(X) = -1|Y = -1]$ , and class imbalance  $b = \Pr[Y = 1] - \Pr[Y = -1]$ , the equation above can be expressed as follows,

$$\begin{aligned} \mu_i = \mathbb{E}[f_i(X)] &= \psi_i \left(\frac{1+b}{2}\right) + (1 - \eta_i) \left(\frac{1-b}{2}\right) - (1 - \psi_i) \left(\frac{1+b}{2}\right) - \eta_i \left(\frac{1-b}{2}\right) \\ &= \psi_i - \eta_i + b(\psi_i + \eta_i - 1) \\ &= 2\delta_i + b(2\pi_i - 1) \end{aligned} \quad (30)$$

where  $\pi_i = (\psi_i + \eta_i)/2$  and  $\delta_i = (\psi_i - \eta_i)/2$ .

Similarly, the population variance of the  $i$ -th classifier is

$$\text{Var}[f_i(X)] = \mathbb{E}[f_i(X)^2] - \mathbb{E}[f_i(X)]^2 = 1 - \mathbb{E}[f_i(X)]^2 = 1 - (2\delta_i + b(2\pi_i - 1))^2. \quad (31)$$

Next, we consider  $\mathbb{E}[f_i(X) \cdot f_j(X)]$ . Under the assumption of independence of errors between different instances and between different classifiers, for  $i \neq j$

$$\begin{aligned} \mathbb{E}[f_i(X) \cdot f_j(X)] &= \Pr[f_i(X) = f_j(X)] - \Pr[f_i(X) = -f_j(X)] \\ &= \left(\frac{1+b}{2}\right) \psi_i \psi_j + \left(\frac{1+b}{2}\right) (1 - \psi_i)(1 - \psi_j) + \left(\frac{1-b}{2}\right) (1 - \eta_i)(1 - \eta_j) + \left(\frac{1-b}{2}\right) \eta_i \eta_j \\ &\quad - \left(\frac{1+b}{2}\right) \psi_i (1 - \psi_j) - \left(\frac{1+b}{2}\right) (1 - \psi_i) \psi_j - \left(\frac{1-b}{2}\right) \eta_i (1 - \eta_j) - \left(\frac{1-b}{2}\right) (1 - \eta_i) \eta_j \end{aligned} \quad (32)$$

Combining the three equations above yields that for  $i \neq j$

$$\begin{aligned} \mathbb{E}[f_i(X) \cdot f_j(X)] - \mathbb{E}[f_i(X)] \cdot \mathbb{E}[f_j(X)] &= (1 - b^2)(\psi_i + \eta_i - 1)(\psi_j + \eta_j - 1) \\ &= (1 - b^2)(2\pi_i - 1)(2\pi_j - 1) \end{aligned} \quad (33)$$

Thus, the entry  $q_{ij}$  of the  $M \times M$  covariance matrix between the  $M$  classifiers is

$$q_{ij} = \begin{cases} 1 - \mu_i^2 & i = j \\ (2\pi_i - 1)(2\pi_j - 1)(1 - b^2) & i \neq j \end{cases} \quad (34)$$

□.

## 2 Direct eigendecomposition of the covariance matrix

*Proof of Lemma 2.2.* Let  $\lambda(Q)$  be the leading eigenvalue of  $Q$  with corresponding unit-norm eigenvector  $\mathbf{w}$ . Let  $\lambda$  be the eigenvalue of the rank-one matrix  $\tilde{Q}$  with corresponding unit-norm eigenvector  $\mathbf{v}$ . First, note that

$$Q = \tilde{Q} + D \quad (35)$$

where  $D$  is a diagonal matrix with entries

$$d_{ii} = 1 - \mu_i^2 - (1 - b^2)(2\pi_i - 1)^2.$$

Hence  $\|D\|_2 = \max_i |d_{ii}| \leq 1$ . It thus readily follows from Weyl's theorem that

$$|\lambda(Q) - \lambda| \leq \|D\|_2 \leq 1. \quad (36)$$

Now we multiply the eigenvector equation  $Q\mathbf{w} = \lambda(Q)\mathbf{w}$  from the left by  $\mathbf{w}^T$ , and insert the relation (35) to obtain that

$$\lambda(Q) = \lambda(\mathbf{w}^T \mathbf{v})^2 + \mathbf{w}^T D \mathbf{w}.$$

The lemma now follows by combining Eq. 36 with the bound  $|\mathbf{w}^T D \mathbf{w}| \leq 1$ .  $\square$ .

## 3 Spectral Meta-Learner

In this section we present the derivation of the Spectral Meta-Learner (SML) as a linearization of the maximum likelihood estimator (MLE) of the vector of true class labels around  $(\psi^*, \eta^*) = (1/2, 1/2)$ .

### 3.1 Maximum Likelihood Estimator (MLE)

Under the assumption of independence between classifiers and instances, given the specificities and sensitivities of the  $M$  classifiers, the overall likelihood of all  $S$  class labels is a product of the likelihood of each individual label. Hence, for each sample  $x_k$  its true class label  $y_k$  can be estimated independently of the other class labels. The MLE  $\hat{y}_k^{(\text{ML})}$  of  $y_k$  is

$$\begin{aligned} \hat{y}_k^{(\text{ML})} &= \operatorname{argmax}_{y_k \in \{1, -1\}} \log \mathfrak{L}(f_1(x_k), \dots, f_M(x_k); y_k) \\ &= \operatorname{argmax}_{y_k} \{\log \mathfrak{L}(f_1(x_k), \dots, f_M(x_k); y_k = 1), \log \mathfrak{L}(f_1(x_k), \dots, f_M(x_k); y_k = -1)\} \\ &= \operatorname{sign}(\log \mathfrak{L}(f_1(x_k), \dots, f_M(x_k); y_k = 1) - \log \mathfrak{L}(f_1(x_k), \dots, f_M(x_k); y_k = -1)) \\ &= \operatorname{sign} \left( \left( \sum_{i|f_i(x_k)=1} \log(\psi_i) + \sum_{i|f_i(x_k)=-1} \log(1 - \psi_i) \right) - \left( \sum_{i|f_i(x_k)=1} \log(1 - \eta_i) + \sum_{i|f_i(x_k)=-1} \log(\eta_i) \right) \right) \\ &= \operatorname{sign} \left( \sum_{i|f_i(x_k)=1} (\log(\psi_i) - \log(1 - \eta_i)) + \sum_{i|f_i(x_k)=-1} (\log(1 - \psi_i) - \log(\eta_i)) \right) \end{aligned}$$

Next, recall that  $f_i(x_k) \in \{-1, 1\}$ . Hence, the conditions  $f_i(x_k) = 1$  and  $f_i(x_k) = -1$  in the two sums above can be replaced by the following two indicator functions,

$$\frac{1 + f_i(x_k)}{2} = \begin{cases} 0 & f_i(x_k) = -1 \\ 1 & f_i(x_k) = 1 \end{cases}$$

and

$$\frac{1 - f_i(x_k)}{2} = \begin{cases} 1 & f_i(x_k) = -1 \\ 0 & f_i(x_k) = 1 \end{cases}$$

Using these indicator functions, we express the MLE as a function of  $\psi_i$  and  $\eta_i$  as follows

$$\begin{aligned} \hat{y}_k^{(\text{ML})} &= \text{sign} \left( \sum_i \frac{1 + f_i(x_k)}{2} (\log(\psi_i) - \log(1 - \eta_i)) + \sum_i \frac{1 - f_i(x_k)}{2} (\log(1 - \psi_i) - \log(\eta_i)) \right) \\ &= \text{sign} \left( \sum_{i=1}^M f_i(x_k) \log \alpha_i + \log \beta_i \right) \end{aligned} \quad (37)$$

where

$$\alpha_i = \frac{\psi_i \eta_i}{(1 - \psi_i)(1 - \eta_i)} \quad \text{and} \quad \beta_i = \frac{\psi_i(1 - \psi_i)}{\eta_i(1 - \eta_i)} \quad (38)$$

### 3.2 The SML: A first-order approximation of the MLE estimator

The maximum likelihood estimate (MLE) of the label  $\hat{y}_k^{\text{ML}}$  of an instance  $x_k$  is given by

$$\hat{y}_k^{(\text{ML})} = \text{sign} \left( \sum_i f_i(x_k) \log \left( \frac{\psi_i \eta_i}{(1 - \psi_i)(1 - \eta_i)} \right) + \log \left( \frac{\psi_i(1 - \psi_i)}{\eta_i(1 - \eta_i)} \right) \right) \quad (39)$$

The first-order Taylor expansion of the MLE, around specificity and sensitivity values  $(\psi_i^*, \eta_i^*)$  is given by

$$\begin{aligned} \hat{y}_k^{(\text{ML})} &= \text{sign} \left( \sum_i f_i(x_k) \log \left( \frac{\psi_i^* \eta_i^*}{(1 - \psi_i^*)(1 - \eta_i^*)} \right) + \log \left( \frac{\psi_i^*(1 - \psi_i^*)}{\eta_i^*(1 - \eta_i^*)} \right) \right. \\ &\quad + f_i(x_k) \left( \frac{(\psi_i - \psi_i^*)}{\psi_i^*} + \frac{(\eta_i - \eta_i^*)}{\eta_i^*} + \frac{(\psi_i - \psi_i^*)}{1 - \psi_i^*} + \frac{(\eta_i - \eta_i^*)}{1 - \eta_i^*} \right) \\ &\quad + \left( \frac{(\psi_i - \psi_i^*)}{\psi_i^*} - \frac{(\eta_i - \eta_i^*)}{\eta_i^*} - \frac{(\psi_i - \psi_i^*)}{1 - \psi_i^*} + \frac{(\eta_i - \eta_i^*)}{1 - \eta_i^*} \right) \Bigg) \\ &\quad + O((\psi_i - \psi_i^*)^2, (\eta_i - \eta_i^*)^2, (\psi_i - \psi_i^*) \cdot (\eta_i - \eta_i^*)) \\ &= \text{sign} \left( \sum_i f_i(x_k) \log \left( \frac{\psi_i^* \eta_i^*}{(1 - \psi_i^*)(1 - \eta_i^*)} \right) + \log \left( \frac{\psi_i^*(1 - \psi_i^*)}{\eta_i^*(1 - \eta_i^*)} \right) \right. \\ &\quad + (\psi_i - \psi_i^*) \frac{f_i(x_k) - (2\psi_i^* - 1)}{\psi_i^*(1 - \psi_i^*)} + (\eta_i - \eta_i^*) \frac{f_i(x_k) + (2\eta_i^* - 1)}{\eta_i^*(1 - \eta_i^*)} \Bigg) \\ &\quad + O((\psi_i - \psi_i^*)^2, (\eta_i - \eta_i^*)^2, (\psi_i - \psi_i^*) \cdot (\eta_i - \eta_i^*)) \end{aligned}$$

At the specific values  $(\psi^*, \eta^*) = (1/2, 1/2)$ , the Taylor expansion above simplifies to

$$\hat{y}_k^{(\text{SML})} = \text{sign} \left( \sum_i f_i(x_k) (\psi_i + \eta_i - 1) \right) = \text{sign} \left( \sum_i f_i(x_k) (2\pi_i - 1) \right) = \text{sign} \left( \sum_i f_i(x_k) v_i \right),$$

where  $v \in \mathbb{R}^M$  is the leading eigenvector of the modified covariance matrix, as described in the main text. We thus call this novel ensemble-classifier the *Spectral Meta-Learner* (SML).

## 4 Covariance between different classifiers in presence of a cartel

*Proof of Lemma 4.1.* As in the proof of Lemma 2.1, we first compute the mean and variance,  $\mu_i = \mathbb{E}[f_i(X)]$  and  $\text{Var}[f_i(X)]$  respectively, of the  $i$ -th classifier. We then use these results to compute the entries of the population covariance matrix,  $q_{ij} = \mathbb{E}[(f_i(X) - \mu_i) \cdot (f_j(X) - \mu_j)]$ .

The mean and variance for  $i \in P$  have been computed in the proof of Lemma 2.1. We now focus on the mean and variance for  $i \in C$ . For brevity, in this section we will use the following notation :

$$\begin{aligned} p_i &= \Pr[f_i(X) = 1|T = 1] & i \in C \\ n_i &= \Pr[f_i(X) = -1|T = -1] & i \in C \\ \psi_c &= \Pr[T = 1|Y = 1] \\ \eta_c &= \Pr[T = -1|Y = -1] \end{aligned} \quad (40)$$

Under the assumption of independence between instances, the population mean for a cartel member  $\mu_i = \mathbb{E}[f_i(X)]$  of the  $i$ -th classifier, with  $i \in C$  is

$$\begin{aligned} \mathbb{E}[f_i(X)] &= \Pr[f_i(X) = 1] - \Pr[f_i(X) = -1] \\ &= \Pr[f_i(X) = 1|T = 1] \Pr[T = 1|Y = 1] \Pr[Y = 1] \\ &\quad + \Pr[f_i(X) = 1|T = -1] \Pr[T = -1|Y = 1] \Pr[Y = 1] \\ &\quad + \Pr[f_i(X) = 1|T = 1] \Pr[T = 1|Y = -1] \Pr[Y = -1] \\ &\quad + \Pr[f_i(X) = 1|T = -1] \Pr[T = -1|Y = -1] \Pr[Y = -1] \\ &\quad - \Pr[f_i(X) = -1|T = 1] \Pr[T = 1|Y = 1] \Pr[Y = 1] \\ &\quad - \Pr[f_i(X) = -1|T = -1] \Pr[T = -1|Y = 1] \Pr[Y = 1] \\ &\quad - \Pr[f_i(X) = -1|T = 1] \Pr[T = 1|Y = -1] \Pr[Y = -1] \\ &\quad - \Pr[f_i(X) = -1|T = -1] \Pr[T = -1|Y = -1] \Pr[Y = -1] \end{aligned} \quad (41)$$

which simplifies to

$$\mathbb{E}[f_i(X)] = b(1 - \psi_c - \eta_c + n_i(\psi_c + \eta_c - 1) + p_i(\psi_c + \eta_c - 1)) + n_i(\psi_c - \eta_c - 1) + p_i(\psi_c - \eta_c + 1) + \eta_c - \psi_c \quad (42)$$

Similarly, as previously shown, the population variance of the  $i$ -th classifier is

$$\text{Var}[f_i(X)] = \mathbb{E}[f_i(X)^2] - \mathbb{E}[f_i(X)]^2 = 1 - \mathbb{E}[f_i(X)]^2 \quad (43)$$

Next, we consider  $\mathbb{E}[f_i(X) \cdot f_j(X)]$ . We remark that the case  $i, j \in P$  was already considered in the proof of Lemma 2.1. Similarly, the case  $i, j \in C$  is a special case of the proof of Lemma 2.1 when the truth is replaced by the cartel's target  $T$ . Thus, for these two cases,

$$\begin{aligned} \mathbb{E}[f_i(X) \cdot f_j(X)] &= (2\pi_i - 1)(2\pi_j - 1)(1 - b^2) & i \neq j, i \in P, j \in P \\ \mathbb{E}[f_i(X) \cdot f_j(X)] &= (2\xi_i - 1)(2\xi_j - 1)(1 - b^2) & i \neq j, i \in C, j \in C \end{aligned} \quad (44)$$

We compute  $\mathbb{E}[f_i(X) \cdot f_j(X)]$  for the cross terms when  $i \in P$  and  $j \in C$ . We define the balanced accuracy  $\pi_c$  of the cartel's target  $T$  with respect to the truth, as well as its sensitivity  $\psi_c$  and specificity  $\eta_c$  with respect to the truth. Under the assumption of independence of errors between different instances and between different classifiers, for  $i \in P, j \in C$

$$\begin{aligned} \mathbb{E}[f_i(X) \cdot f_j(X)] &= \Pr[f_i(X) = f_j(X)] - \Pr[f_i(X) = -f_j(X)] \\ &= ((2\psi_i - 1)((1 - 2n_j)(1 - \psi_c) - (1 - 2p_j)\psi_c))(1 + b)/2 \\ &\quad + ((2\eta_i - 1)((1 - 2p_j)(1 - \eta_c) - (1 - 2n_j)\eta_c))(1 - b)/2 \end{aligned} \quad (45)$$

Combining the three equations above yields that for  $i \in P, j \in C$

$$\begin{aligned} \mathbb{E}[f_i(X) \cdot f_j(X)] - \mathbb{E}[f_i(X)] \cdot \mathbb{E}[f_j(X)] &= (1 - b^2)(\psi_i + \eta_i - 1)(\psi_c + \eta_c - 1)(n_j + p_j - 1) \\ &= (1 - b^2)(2\pi_i - 1)(2\pi_c - 1)(2\xi_j - 1) \end{aligned} \quad (46)$$



Thus, the entry  $q_{ij}$  of the  $M \times M$  covariance matrix between the  $M$  classifiers is

$$q_{ij} = \begin{cases} 1 - \mu_i^2 & i = j \\ (2\pi_i - 1)(2\pi_j - 1)(1 - b^2) & i \neq j, i \in P, j \in P \\ (2\pi_i - 1)(2\pi_c - 1)(2\xi_j - 1)(1 - b^2) & i \in P, j \in C \\ (2\xi_i - 1)(2\xi_j - 1)(1 - b^2) & i \neq j, i \in C, j \in C \end{cases} \quad (47)$$

□.

From Lemma 4.1 it follows that the matrix  $Q$  can be written as a block matrix

$$Q = \left[ \begin{array}{c|c} Q_P & Q_{PC} \\ \hline Q_{PC}^T & Q_C \end{array} \right], \quad (48)$$

where both  $Q_P$  and  $Q_C$  are rank one, and  $Q_{PC}$  represents the interaction between classifiers in  $P$  with the classifiers in  $C$ .

## 5 Matrix rank and eigendecomposition of the off-diagonal elements of the covariance between different classifiers in presence of a cartel

*Proof of Theorem 4.2.* In the present proof, we simplify the notation using the following convenient change of variables:

$$\begin{aligned} \rho_i &= 2\pi_i - 1 \\ \tau_i &= 2\xi_i - 1 \\ u &= (1 - b^2); \end{aligned} \quad (49)$$

Suppose that the off-diagonal terms of the symmetric real-valued covariance matrix  $Q$  correspond to a rank-two matrix, then we can write them as a linear combination of the outer products of two orthogonal vectors,  $\mathbf{e}_1$  and  $\mathbf{e}_2$  (the eigenvectors):

$$q_{ij} = \lambda_1 e_{1i} e_{1j} + \lambda_2 e_{2i} e_{2j}. \quad (50)$$

We parametrize these eigenvectors in a block form.

$$\lambda_1 e_{1i} = \begin{cases} u \cdot a_{11} \rho_i & i \in P \\ u \cdot a_{21} \tau_i & i \in C \end{cases} \quad (51)$$

$$\lambda_2 e_{2i} = \begin{cases} u \cdot a_{12} \rho_i & i \in P \\ u \cdot a_{22} \tau_i & i \in C \end{cases} \quad (52)$$

Since the eigenvectors are orthogonal it follows that

$$\sum_{i \in P} a_{11} \rho_i a_{12} \rho_i + \sum_{j \in C} a_{21} \tau_j a_{22} \tau_j = 0 \quad (53)$$

Let us rewrite the matrix in Eq. 50 in a block matrix form by plugging the block eigenvectors defined in Eq. 51 and Eq. 52:

$$\begin{cases} u\rho_i\rho_j = u \cdot a_{11}\rho_i \cdot a_{11}\rho_j + u \cdot a_{12}\rho_i \cdot a_{12}\rho_j & i \in P, j \in P \\ u\rho_i\rho_c\tau_j = u \cdot a_{11}\rho_i \cdot a_{21}\tau_j + u \cdot a_{12}\rho_i \cdot a_{22}\tau_j & i \in P, j \in C \\ u\tau_i\tau_j = u \cdot a_{21}\tau_i \cdot a_{21}\tau_j + u \cdot a_{22}\tau_i \cdot a_{22}\tau_j & i \in C, j \in C \end{cases} \quad (54)$$

Hence, if  $a_{11}$ ,  $a_{12}$ ,  $a_{21}$  and  $a_{22}$  satisfy the following set of equations

$$\begin{cases} a_{11}^2 + a_{12}^2 &= 1 \\ a_{11}a_{21} + a_{12}a_{22} &= \rho_c \\ a_{21}^2 + a_{22}^2 &= 1 \\ (a_{11}a_{12})/(a_{21}a_{22}) &= -(\sum_{j \in C} \tau_j^2)/(\sum_{i \in P} \rho_i^2) \end{cases} \quad (55)$$

then the left hand side of Eq. 50 is a rank-2 matrix.

Following a change of variables, with  $a_{11} = \cos \alpha$ ,  $a_{12} = \sin \alpha$ ,  $a_{21} = \sin \beta$ , and  $a_{22} = \cos \beta$ , Eq. 55 reduces to

$$\begin{cases} \cos \alpha \sin \beta + \sin \alpha \cos \beta &= \rho_c \\ (\cos \alpha \sin \alpha)/(\sin \beta \cos \beta) &= -(\sum_{j \in C} \tau_j^2)/(\sum_{i \in P} \rho_i^2) \end{cases} \quad (56)$$

Next, we show that the system in Eq. 56 is determined and that it has a unique solution up to a rotation.

We define  $k_1 = \rho_c$  and  $k_2 = (\sum_{j \in C} \tau_j^2)/(\sum_{i \in P} \rho_i^2)$ , and simplify the system in Eq. 56

$$\begin{cases} \sin(\alpha + \beta) &= k_1 \\ \sin(2\alpha)/\sin(2\beta) &= -k_2 \end{cases} \quad (57)$$

Defining  $\delta = \alpha + \beta$ , we obtain

$$\sin \delta = k_1 \quad (58)$$

$$\cos \delta = \sqrt{1 - k_1^2} \quad (59)$$

$$\sin(2\delta) = 2k_1\sqrt{1 - k_1^2} \quad (60)$$

$$\cos(2\delta) = 1 - 2k_1^2 \quad (61)$$

We rewrite Eq. 57 as follows, solving for  $\alpha$

$$\sin(2\alpha) + k_2 \sin(2\delta - 2\alpha) = 0 \quad (62)$$

$$\sin(2\alpha) + k_2 \sin(2\delta) \cos(2\alpha) - k_2 \cos(2\delta) \sin(2\alpha) = 0 \quad (63)$$

$$\tan(2\alpha) = -\frac{k_2 \sin(2\delta)}{1 - k_2 \cos(2\delta)} \quad (64)$$

$$\alpha = \frac{1}{2} \arctan \left( \frac{k_1 k_2}{k_2(1 - 2k_1^2) - 1} \right), \quad (65)$$

and, similarly, solving for  $\beta$

$$\sin(2\delta - 2\beta) + k_2 \sin(2\beta) = 0 \quad (66)$$

$$\sin(2\delta) \cos(2\beta) + \cos(2\delta) \sin(2\beta) + k_2 \sin(2\beta) = 0 \quad (67)$$

$$\tan(2\beta) = -\frac{\sin(2\delta)}{k_2 - \cos(2\delta)} \quad (68)$$

$$\beta = \frac{1}{2} \arctan \left( \frac{2k_1\sqrt{1 - k_1^2}}{1 - k_2 - 2k_1^2} \right) \quad (69)$$

These solutions of  $\alpha$  and  $\beta$  are unique up to a rotation with periodicity  $\frac{\pi}{2}$ . We recall that  $a_{11} = \cos \alpha$ ,  $a_{12} = \sin \alpha$ ,  $a_{21} = \sin \beta$ , and  $a_{22} = \cos \beta$ . The eigenvectors and their respective eigenvalues can be easily derived by back-substitution in Eq. 51 and Eq. 52.

The system in Eq. 57 is therefore a determined system of two equations in two variables. It is thus possible to express the off-diagonal terms of the matrix  $Q$  as the linear combination of the outer products of two orthogonal vectors  $\mathbf{e}_1$  and  $\mathbf{e}_2$ , defined by the angles  $\alpha$  and  $\beta$ . Therefore, it follows that the off-diagonal elements of the matrix  $Q$  correspond to a symmetric real-valued rank-two matrix whose eigenvectors are the two orthogonal vectors  $\mathbf{e}_1$  and  $\mathbf{e}_2$ .

Importantly, the classifiers belonging to the  $P$  group lay on a line with angle  $\alpha$  relative to the eigenvector  $e_1$ . The classifiers in the cartel, lay on a line with angle  $\beta$  relative to the eigenvector  $e_2$ .  $\square$ .

## 6 Simulations and benchmarks

The following section describes how we generated the simulated data and how we performed the benchmarks. For each component of the simulations we also provide pseudo-code

### 6.1 Simulated data: Ensembles of statistically independent predictions

We generated ensembles of statistically independent predictions using previously described random detector with fixed balanced accuracy (RDFBA) [24]. A generic RDFBA predictor with pre-determined balanced accuracy equal to  $\pi$  is denoted by  $\text{RDFBA}(\pi)$ .  $\text{RDFBA}(\pi)$  predictions are used to simulate predictions from independent classifiers. RDFBAs are constructed such that their balanced accuracy is equal to  $\pi$ , although the sensitivity  $\psi$  and specificity  $\eta$  may be different for equal choices of  $\pi$ .

Following the standard machine-learning notation,  $P$  is the number of positives, i.e. the number of instances whose true class label is  $+1$ ;  $N$  is the number of negatives, i.e. the number of instances whose true class label is  $-1$ ;  $FP$  is the number of false positives, i.e. the number of negatives that have been mistakenly predicted as positives;  $FN$  is the number of false negatives, i.e. the number of positives that have been mistakenly predicted as negatives. Thus, an  $\text{RDFBA}(\pi)$  prediction is constructed from the ground truth vector  $y$  as follows:

1. the entries of prediction vector  $f(x)$  are initialized with the corresponding entries in the ground truth vector  $y$ .
2. Under the constraint that  $FN = (2 - 2\pi - FP/N) \cdot P$  is an integer, a random integer number  $FP$  is drawn with uniform probability from  $[0, N]$ .
3.  $FP$  instances in  $f(x)$ , whose true label is  $-1$ , are assigned the wrong class label,  $+1$ .
4.  $FN$  instances in  $f(x)$ , whose true label is  $+1$ , are assigned the wrong class label,  $-1$ .

The advantage of using RDFBA predictors is that each prediction satisfies the assumption of independence between predictors.

In our simulations, we used  $P = N = 300$ , and  $\pi \in [0.3, 0.8]$ .

### 6.2 Simulated data: Ensembles of uncorrelated predictions with a small cartel of strongly correlated predictors

In order to generate datasets of uncorrelated predictions where a small ( $r \cdot M$  predictors) cartel was present, we first generated an ensemble  $P$  of  $(1 - r)M$  independent predictions as described above for

the ground truth vector  $\mathbf{y}$ . Then, we constructed the cartel’s target vector  $\mathbf{c}$ , i.e. a vector alternative to the truth that is supported by the predictions in the cartel. The vector  $c$  is constructed as an RDFBA prediction with balanced accuracy  $\pi_c$ . For this vector  $c$  we constructed an ensemble  $C$  of independent predictions similarly to the procedure described for the statistically independent predictions with the only difference that the balanced accuracies of all members of the cartel relative to the cartel’s target was set to be equal to 0.7. The dataset is obtained by the union of the two ensembles of predictions,  $P$  and  $C$ . In our simulations we used  $\pi_c = 0.5$  thus obtaining a cartel’s target that is orthogonal to the ground truth.

### 6.3 Real data: Ensembles of predictions from standard machine-learning classifiers

To generate ensemble of predictions from standard machine-learning classifiers on real data, we trained the classifiers on partially overlapping training data and collected their predictions obtained on the same testing data, which was independent from all the training data. In detail, from each dataset we sampled 600 instances (or all the instances if less than 600 were available), half of which (up to 300) were used for testing. Independently for each classifier, we selected a random subset comprising of 90% of the instances reserved for training and used this subset as a ”private” training set. The purpose of this procedure was to produce training data that was slightly different between the different classifiers, allowing, at the same time, to have a significantly large number of training samples even in the smaller datasets. We chose to use at most 600 instances to reduce computational time. To determine the empirical distribution of performances of each classifier and of the ensemble approaches discussed in the manuscript, for each dataset we repeated this procedure 1000 times.

## 7 Supplemental Tables

Table S1: **Summary of the datasets from the UCI repository [23].**

<b>Dataset</b>	<b>Instances</b>	<b>Features</b>	<b>Class</b>	<b>Reference</b>
Spambase data - SD	4601	57	spam/not spam	[23]
Yale breast cancer dataset - YBC	650	6	nodal status	[22]
Wisconsin breast cancer dataset - WBC	699	10	benign/malignant	[26]
Parkinson data - PD	197	23	affected/unaffected	[27]
MAGIC Gamma Telescope - MGT	19020	11	signal/background	[28]
Ionosphere data - ID	351	34	Good return/Bad return	[29]
Mammographic masses - MM	961	6	disease severity (2 classes)	[30]
Abalone data - AD	4177	8	male/female	[31]

Table S2: **Summary of the machine learning classifiers from Weka [32].**

<b>classifier/meta-learner</b>	<b>Weka class</b>
KNN (k=1, odd)	lazy/IBk
KNN (k=2, even)	lazy/IBk
KNN (k=5)	lazy/IBk
k-Star	lazy/KStar
DecisionStump	trees/DecisionStump
J48	trees/J48
REPTree	trees/REPTree
JRip	Rules/JRip
LMT	trees/LMT
LWL	lazy/LWL
Regularized Logistic regression	functions/Logistic
Logistic regression	functions/SimpleLogistic
Sequential Minimal Optimization	function/SMO
NaiveBayes	bayes/NaiveBayes
M5P	rules/M5P
OneR	rules/OneR
PART	rules/PART
RandomForest (n=10 trees)	trees/RandomForest
RandomForest (n=20 trees)	trees/RandomForest
Multilayer Perceptron	functions/MultilayerPerceptron
Voted Perceptron	functions/VotedPerceptron
SGD	functions/SGD
Voting	meta/Vote
Stacking	meta/Stacking
AdaBoost + NaiveBayes	meta/AdaBoostM1
AdaBoost + Logistic Regression	meta/AdaBoostM1
AdaBoost + J48	meta/AdaBoostM1
Bagging + REPTree	meta/Bagging
Bagging + RandomTree	meta/Bagging
Bagging + RandomForest	meta/Bagging
LogitBoost + ZeroR	meta/LogitBoost
LogitBoost + KNN	meta/LogitBoost
LogitBoost + DecisionStump	meta/LogitBoost

## 8 Supplemental Figures

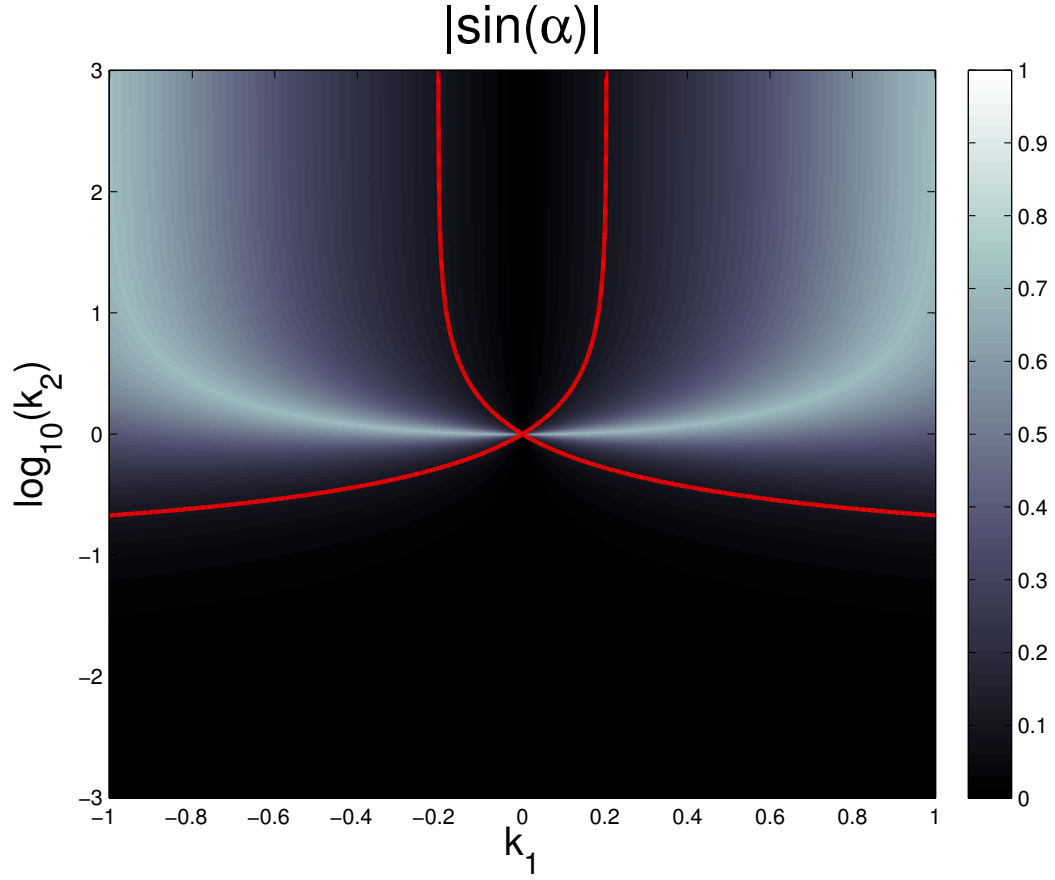


Figure S4: The heatmap shows the absolute value of the angle between the truth and the eigenvector  $e_1$ , on which the SML prediction is based. The dark area between the two red lines graphically shows the relationship between  $k_1$  and  $k_2$  such that  $|\alpha| \leq 6^\circ$ . The figure shows that SML is robust to cartels: when  $\alpha \approx 0$ , the honest classifiers lie approximatively on the eigenvector  $e_1$ .

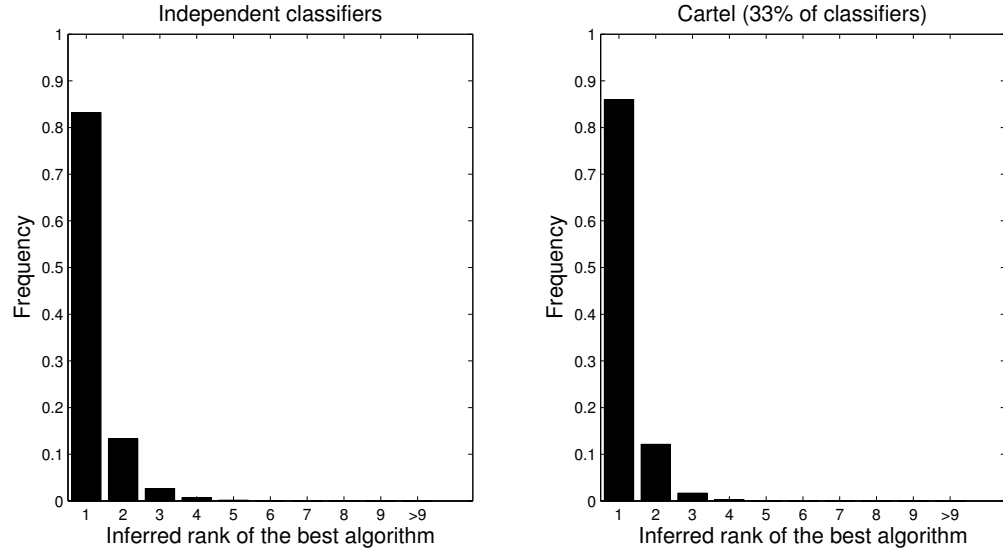


Figure S5: The largest entry in the leading eigenvector often corresponds to the best classifier in the ensemble. In the plots, each bar represent the empirical probability that the entry in the leading eigenvector corresponding to best classifier attained a specific rank.



## References

- [1] Aristotle Politics
- [2] A. P. Dawid and A. M. Skene. Maximum Likelihood Estimation of Observer Error-rates using the EM algorithm. *Appl.Statist.*, 28, No. 1, pp. 20-28, 1979.
- [3] Plato The Republic
- [4] H. A. Linstone and M. Turoff The Delphi Method: Techniques and Applications *Addison-Wesley*, ISBN 978-0-201-04294-8
- [5] A. Timmermann *Handbook of economic forecasting, Chapter 4 - Forecast combinations*. Elsevier, 2006
- [6] D. R. Karger, S. Oh and D. Shah. Budget-optimal crowdsourcing using low-rank matrix approximations, *49th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, 284–291, 2011.
- [7] D. R. Karger, S. Oh and D. Shah. Iterative Learning for Reliable Crowdsourcing Systems *NIPS* , 2011
- [8] J. Whitehill, P. Ruvolo., T. Wu, J. Bergsma and J. Movellan. Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. *Advances in Neural Information Processing Systems*, 22:2035–2043, 2009.
- [9] P. Smyth, U. Fayyad, M. Burl, P. Perona and P. Baldi Inferring ground truth from subjective labelling of venus images. *Advances in Neural Information Processing Systems* 7, pp. 1085–1092, 1995.
- [10] V. S. Sheng, F. Provost, and P. G. Ipeirotis. Get another label? Improving data quality and data mining using multiple, noisy labelers. *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 614–622, 2008.
- [11] P. Welinder, S. Branson, S. Belongie, and P. Perona. The multidimensional wisdom of crowds. *Advances in Neural Information Processing Systems*, 23, pp. 2424–2432, 2010.
- [12] Y. Yan, R. Rosales, G. Fung, M. Schmidt, G. Hermosillo, L. Bogoni, L. Moy, and J. Dy Modeling annotator expertise: Learning when everybody knows a bit of something. *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics (AISTATS 2010)*, pp. 932–939, 2010.
- [13] V. C. Raykar, S. Yu, L. H. Zhao, G. H. Valadez, C. Florin, L. Bogoni, and L. Moy Learning from crowds *Journal of Machine Learning Research*, 11, pp.1297–1322, 2010.
- [14] R. Jin and Z. Ghahramani Learning with Multiple Labels. *Advances in Neural Information Processing Systems 15: Proceedings of the 2002 Conference*.
- [15] S. L. Lauritzen The EM algorithm for graphical association models with missing data. *Computational Statistics & Data Analysis*, 19, No.2, pp. 191-201, 1995
- [16] R. Snow, B. O’Connor, D. Jurafsky and A. Y. Ng Cheap and fast—but is it good?: evaluating non-expert annotations for natural language tasks. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2008, pp. 254–263

- [17] S. D. Walter and L. M. Irwig. Estimation of test error rates, disease prevalence and relative risk from misclassified data: a review *Journal of Clinical Epidemiology*, 41, Issue 9, 1988, pp. 923-937, ISSN 0895-4356, 10.1016/0895-4356(88)90110-2.
- [18] T.G. Dietterich, Ensemble methods in machine learning, Multiple Classifier Systems, 2000.
- [19] E.J. Candès, and B. Recht. Exact matrix completion via convex optimization. *Foundations of Computational Mathematics*, 9(6):717–772, 2009.
- [20] I. Witten, E. Frank, and M. Hall. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2011.
- [21] H. Cheng and K.W. Shui *Is there an optimal forecast combination?*. IEPR Working paper, September 2011
- [22] F. Parisi, A. M. González, Y. Nadler, R. L. Camp, D. L. Rimm, H. M. Kluger, and Y. Kluger. Benefits of biomarker selection and clinico-pathological covariate inclusion in breast cancer prognostic models. *Breast Cancer Res*, 12(5):R66, Sep 2010.
- [23] A. Frank and A. Asuncion. UCI machine learning repository. Irvine, CA: University of California,
- [24] F. Strino, F. Parisi, and Y. Kluger. VDA, a method of choosing a better algorithm with fewer validations. *PLoS ONE*, 6(10):e26074, 2011.
- [25] R. McGill, J.W. Tukey and W.A. Larsen *Variations of Box Plots* The American Statistician 32 (1):12-16, 1978
- [26] W. H. Wolberg and O. L. Mangasarian. Multisurface method of pattern separation for medical diagnosis applied to breast cytology. *Proc Natl Acad Sci U S A*, 87(23):9193–6, Dec 1990.
- [27] M. A. Little, P. E. McSharry, S. J. Roberts, D. A. E. Costello, and I. M. Moroz. Exploiting nonlinear recurrence and fractal scaling properties voice disorder detection. *Biomed Eng Online*, 6:23, 2007.
- [28] D. Heck, J. Knapp, J. Capdevielle, G. Schatz, and T. Thouw. Report FZKA 6019, Forschungszentrum Karlsruhe, 1998. Technical report, 1986.
- [29] V. Sigillito, S. Wing, L. Hutton, and K. Baker. Classification of radar returns from the ionosphere using neural networks. *Johns Hopkins APL Technical Digest*, 10(3):262–266, 1989.
- [30] M. Elter, R. Schulz-Wendtland, and T. Wittenberg. The classification of breast cancer biopsy outcomes using two approaches that both emphasize an intelligible decision process. *Med Phys*, 34(11):4164–72, Nov 2007.
- [31] P. McShane and J. Reyn. Small-scale spatial variation in growth, size at maturity, and yield-and egg-per-recruit relations in the new zealand Abalone *Haliotis* *New Zealand Journal of Marine and Freshwater Research*, 29(4):603–612, 1995.
- [32] I. Witten, E. Frank, and M. Hall. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2011.